# **C#: Remove duplicates** from an Array

**using System;**

**using System.Linq;**

**using System.Collections.Generic;**

**public class Program**

**{**

**public static void Main()**

**{**

**string[] values = { "abc", "xyz","abc", "def", "ghi", "asdf", "ghi","xd", "abc" };**

**values = new HashSet<string>(values).ToArray();**

*//values = values.Distinct().ToArray();* ***//using******linq***

**foreach(var value in values){**

**Console.WriteLine(value);**

**}**

**}**

**}**

Link : <https://www.dotnetperls.com/duplicates>

# JavaScript: Remove Duplicates from an Array

There are multiple ways to remove duplicates from an array.The simplest approach (in my opinion) is to use the [Set](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Set) object which lets you store **unique values** of any type. In other words, Set will automatically remove duplicates for us.

**const** names **=** ['John', 'Paul', 'George', 'Ringo', 'John'];

**let** unique **=** [...**new** Set(names)];console.log(unique); *// 'John', 'Paul', 'George', 'Ringo'*

Another option is to use [filter()](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/filter).

**const** names **=** ['John', 'Paul', 'George', 'Ringo', 'John'];

**let** x **=** (names) **=>** names.filter((v,i) **=>** names.indexOf(v) **===** i)x(names); *// 'John', 'Paul', 'George', 'Ringo'*

And finally we can use forEach().

**const** names **=** ['John', 'Paul', 'George', 'Ringo', 'John'];

**function** removeDups(names) {

**let** unique **=** {};

names.forEach(**function**(i) {

**if**(**!**unique[i]) {

unique[i] **=** **true**;

}

});

**return** Object.keys(unique);}

removeDups(names); *// // 'John', 'Paul', 'George', 'Ringo'*